3D Trajectory Estimation of Simulated Fruit Flies

Junli Tao
University of Auckland
Auckland, New Zealand
jtao076@aucklanduni.ac.nz

Benjamin Risse
University of Münster
Münster, Germany
b.risse@wwu.de

Xiaoyi Jiang
University of Münster
Münster, Germany
xjiang@uni-muenster.de

Reinhard Klette
University of Auckland
Auckland, New Zealand
r.klette@auckland.ac.nz

ABSTRACT

This paper addresses 3D trajectory estimation of simulated fruit flies assuming a time-synchronized and calibrated 3-camera system. Because the objects have almost the same appearance, both stereo matching and temporal tracking are challenging. In this paper, a third camera is introduced to verify matching and tracking results based on epipolar geometry and projection consistency. This reduces the ambiguity, fetches missed matches and corrects incorrect matches during tracking. Since matching and tracking affect each other, we process both in interaction instead of separately. Unscented Kalman filters are adopted to track objects by modelling motion information, as no distinguishing appearance features are available.

Categories and Subject Descriptors

I.4.8 [**Computing Methodologies**]: Image Processing and Computer Vision—*Scene Analysis, Tracking*

Keywords

Drosophila melanogaster, fruit flies, stereo matching, 3D tracking, unscented Kalman filter

1. INTRODUCTION

Numerous research is based on visual analysis of trajectories of small irregularly moving insects; see, for example, [1, 5]. *Drosophila melanogaster* (fruit flies) is one of the most important model organism in biology since it is a powerful model system for questions regarding the neuronal, sensory and genetic foundations. Approaches dealing with crawling flies are common praxis, thus 2D tracking techniques have been established as a standard method for behavioural studies [19]. Since 2D approaches are limited to crawling animals, the flight ability is suppressed by various means. Being restricted in their movement, the animals often show different and partly unnatural behaviours [6]. Furthermore, phenotypic (i.e. visually perceivable) effects on flight behaviour will be completely missed. Work on 3D tracking for freely flying fruit flies is still in its initial stages because the automatic reconstruction of 3D trajectories for these objects is a challenging task.

To obtain 3D trajectories, two subprocesses are involved, stereo matching on different views for reconstructing 3D coordinates for objects, and temporal tracking within each view to identify object correspondences over time. As the objects have similar appearance, extensively used stereo matching methods [4, 12] fail because they are dependent on local texture information. Without having distinguishing appearance information available, motion information can only be adopted by temporal tracking. However, this is insufficient due to occlusions and identity ambiguities, especially in (what we call) *crowded scenes* of large numbers of simulated fruit flies.

In this paper, we propose to solve stereo matching and temporal tracking in mutual support, and to introduce a third camera to estimate 3D trajectories of simulated fruit flies. With the introduction of a third camera, the matched stereo point-pair from any two cameras can be verified with the third image. Projections of one object into three time-synchronized images satisfy the epipolar constraint and *projection consistency*. The corresponding stereo point lies on the epipolar line. The three projections of 3D coordinates are consistent with each other. Instead of solving stereo matching and temporal tracking consequently [23], we deal with them alternatively. We require exhaustive matching of corresponding projections in the first triple of frames only. An *unscented Kalman filter* (UKF), modelling motion information, is then used to track the projections in 2D image planes. Using the epipolar constraint and by checking projection consistency, tracking results are verified to obtain a valid projection correspondence for the current three-camera frame; no search is involved. In case of an incorrect match, ambiguity or a missed match occur during tracking. We can solve these issues and obtain valid corresponding projections satisfying the epipolar geometry constraint and projection consistency.

2. RELATED WORK

The first attempt to track *Drosophila melanogaster* was made by Buelthoff et al. in 1980 and was based on a single camera in combination with a mirror leaning forward at 45

IVCNZ '12 Dunedin, NZ

degree over the tracking chamber [2]. 20 years later Fry et al. published an approach to track flying insects applying modern optics and hardware such as pan-tilt cameras [6]. Furthermore, Tammero et al. used calibrated stereo infrared cameras to track a fly in a circular arena [21]. All approaches have in common that it is only possible to track a single target at a time (no temporal tracking) and therefore reduces the NP-hard complexity of the multi-index assignment problem. Others tried to avoid expensive multi-camera multi-target 3D tracking by tracking in two dimensions (no stereo matching) only [3, 10, 15, 16]. Since all above mentioned 2D tracking approaches prevent the insects from flying, they are inappropriate for general movement studies. Thus, Drosophila larvae are used for large scale screening studies since several phenotypic effects can also be observed at this stage [7, 13, 18]. However, it is impossible to detect characteristics of a free flight of an adult Drosophila in these approaches.

Methods for estimating 3D trajectories are typically based on stereo matching and temporal tracking; consequently they require two or more cameras. In [5, 23], 2D tracks are calculated in the image plane, and then matched between cameras to reconstruct 3D trajectories. Alternatively, in [8, 17, 20], stereo matching is used to reconstruct 3D coordinates, then followed by tracking of 3D points to obtain 3D trajectories. However, these methods are vulnerable to either stereo or tracking ambiguities. [24] handles stereo matching and tracking simultaneously by minimizing a cost function related to the epipolar constraint, kinetic coherency, and observation matches. This method deals with the ambiguity, but it is expensive in terms of computation time.

In [8, 20], more than two cameras are used for the 3D tracking task. [8] adopts three cameras for reconstructing a 3-dimensional hull for the flies, and then tracks the hull with an *extended Kalman filter* (EKF). [20] employs up to eleven cameras for realtime trajectory estimation. But both methods did not consider to use an extra camera for stereo matching verification and further reduction of ambiguity affects in tracking.

In this paper, with an insight into interaction between stereo matching and temporal tracking, we solve both analysis problems in mutual interaction. Having motion modelled and estimated by an UKF, an exhaustive search is only conducted for the first three synchronized frames, and search results are then propagated through the sequence.

3. PROPOSED ALGORITHM

The proposed algorithm expects inputs to be time-synchronized sequences from three arbitrary directed and calibrated cameras, denoted by Camera 1, Camera 2, and Camera 3. Corresponding images are represented by $\mathcal{I}_t^i$, where $i = 1, 2, 3$, and t is the time index.

Assume that detection in the current frame has been solved, prior to matching and tracking. Positions are denoted by $\mathbf{d}_{n_i}^i = (x, y)$, meaning the n_ith object detected in $\mathcal{I}^i$, where (x, y) is the image coordinate of the object centroid, and $n_i = 1, 2, \ldots, N_i$. The N_i values may differ due to occlusion. As only detections in the current frame are considered, the time index is neglected for simplicity.

Consider *triplets* of image points (i.e. for the three projections) corresponding to one 3D coordinate, denoted by $(\mathbf{d}_{n_1}^1, \mathbf{d}_{n_2}^2, \mathbf{d}_{n_3}^3)$. Initial triplets are found by matching and verification in the first three images (see Section 3.1). This exhaustive search is only needed for the first stereo-frame.

A family of UKFs, one UKF for each detection, is initialized including partially occluded ones. Then, objects are tracked in the 2D image plane (Section 3.2). The tracked triplets (obtained from three UKFs separately) are verified to decide whether a triplet is valid. An invalid triplet means that at least one tracking result is wrong. Further validation is done to find and correct incorrect tracking results. If one of the three tracks is missing, the other two can be used to find the missed match. When two detections are close to each other in $\mathcal{I}^i$, an ambiguity for data association occurs. With corresponding detections in the other two images, projection consistency (details in Section 3.1) can solve the ambiguity to some degree (depending on how close the two detections are).

3.1 Stereo Matching and Verification

Given three sets of detected object centres $\mathbf{D}^i$, triplets are supposed to be identified in them. As no distinguishable appearance features are available, epipolar geometry [9] and projection consistency are used to match and verify the stereo triplets. Epipolar geometry provides constraints for matching. With point $\mathbf{d}_{n_1}^1$ in $\mathcal{I}_t^1$, the matched point $\mathbf{d}_{n_2}^2$ in $\mathcal{I}_t^2$ should be located on the epipolar line. Projection consistency means that the projection of a 3D coordinate, reconstructed by two points $(\mathbf{d}_{n_1}^1, \mathbf{d}_{n_2}^2)$ of a triplet, should be consistent with the third point $(\mathbf{d}_{n_3}^3)$.

With point $\mathbf{d}_{n_i}^i$, the corresponding epipolar line in $\mathcal{I}_t^j$ is obtained by

$$\begin{aligned} \mathbf{l}_{n_i}^j &= \mathbf{F}\mathbf{d}_{n_i}^i \\ \mathbf{F} &= [\mathbf{K}^j\mathbf{t}^{ij}]_\times \mathbf{K}^j\mathbf{R}^{ij}(\mathbf{K}^i)^{-1} \end{aligned}$$

where $\mathbf{K}^i, \mathbf{K}^j$ are the camera matrixes, $\mathbf{R}^{ij}, \mathbf{t}^{ij}$ are the rotation matrix and translation from Camera i to Camera j, and $\mathbf{F}$ is the *fundamental matrix*; for details see [9]. Detected points $\mathbf{d}^j$, lying on the epipolar line, are matched to $\mathbf{d}_{n_i}^i$. Thus, a set of 3D coordinates, $\mathbf{p}_m$, $m = 1, \ldots, M$, is reconstructed by those stereo pairs $(\mathbf{d}_{n_i}^i, \mathbf{d}_m^j)$.

Verification is applied to choose a valid match for $\mathbf{d}_{n_i}^i$ with projection consistency. The 3D coordinates $\mathbf{p}_m$ are projected into the third image $\mathcal{I}^h$ applying the pinhole camera model, thus obtaining image points $\mathbf{q}_m^h$. The valid match $\mathbf{d}_*^j$ is obtained by

$$\begin{aligned} \mathbf{q}_*^h &= \min_m dist(\mathbf{q}_m^h, \mathbf{d}_*^h) \\ \mathbf{d}_*^h &= \min_{n_h}(dist(\mathbf{q}_m^h, \mathbf{d}_{n_h}^h) + (\mathbf{l}_{n_i}^h)^\mathbf{T}\mathbf{d}_{n_h}^h) \end{aligned}$$

where $dist(a, b)$ means calculating the Euclidean distance, $\mathbf{d}_*^h$ is the closest detection satisfying epipolar constraints to $\mathbf{q}_m^h$, and $\mathbf{q}_*^h$ denotes the most consistent projection, which corresponds to $\mathbf{p}_*$ and also to $\mathbf{d}_*^j$. Thus, we found the triplet $(\mathbf{d}_{n_i}^i, \mathbf{d}_*^j, \mathbf{d}_*^h)$.

3.2 2D Tracking

An UKF is adopted for 2D tracking by modelling the position and motion information in the process model. One UKF is applied for one object. As an unknown number of objects appears and disappears in the scene, a state space for the whole process would expand significantly when dealing with crowded scenes. This would slow down the process dramatically. Thus, we define the state to be $\mathbf{x} = (x, y, v_x, v_y)^T$, $\mathbf{x}^p = (x, y)$. A constant velocity with Gaussian distributed noise acceleration $\mathbf{n_a} \in N(0, \Sigma_{n_a})$ assumption is made for

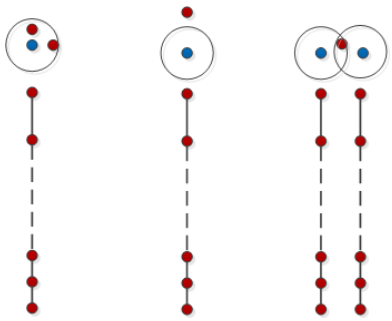

Figure 1: Tracking ambiguity. Blue dots denote prediction, red dots denote detection, circles denote the region with radius (i.e. threshold) τ_1

the process model:

$$\dot{x} = (v_x + n_{ax}\Delta t)\Delta t, \quad \dot{v_x} = n_{ax}\Delta t$$
$$\dot{y} = (v_y + n_{ay}\Delta t)\Delta t, \quad \dot{v_y} = n_{ay}\Delta t$$

Here, Δt is the time interval between subsequent frames. With observation $\mathbf{y}_t$ at time t, the state can be estimated with prediction $\mathbf{x}_{t|t-1}$ (obtained by the process model via the previous state), estimated observation $\mathbf{y}_{t|t-1}$, and Kalman gain $\mathcal{K}_t$; for details see [22]:

$$\mathbf{x}_{t|t} \quad = \quad \mathbf{x}_{t|t-1} + \mathcal{K}_t(\mathbf{y_t} - \mathbf{y_{t|t-1}})$$

Since object movements are continuous, the estimated velocity can be used as a cue to localize the search area for finding the match object in the current frame. For each 2D track, the possible location of the object in the current frame is predicted by the process model with the previous state $\mathbf{x}_{t-1|t-1}$. This location $\mathbf{x}_{t|t-1}$ is used as a reference for searching potential match candidates in the current frame.

If $dist(\mathbf{x}^p_{t|t-1}, \mathbf{d}_n)$ is smaller than a given threshold τ_1, $\mathbf{d}_n$ is taken as candidate detection for the 2D trajectory at time t. As no appearance feature is available for further verification, ambiguities exist in tracking as shown in Figure 1. Ambiguities can be as follows:

- One detection is matched with several trajectories.
- One trajectory is matched with several detections.
- Some trajectories fail to match with any detection.
- Some detections fail to match with any trajectory.

3.3 3D Trajectory Reconstruction

We propose to operate stereo matching and temporal tracking alternatively, with insights into interactions between them. We only use exhaustive search for triplets in the first stereo frame instead of in each frame, as in [24]. Tracking ambiguities are handled by the proposed strategy as well.

Figure 2 shows the work flow of the proposed strategy. The triplets in the first three frames are found by exhaustive search. Then, trackers (i.e. UKFs) are initialized and predictions are obtained. For each tracked triplet, the missed matches, ambiguity, verification, and correction are done with respect to the needs. After updating all trackers, new triplets (i.e. not yet matched to any track) are found and handled. Then we do the iteration again with the new frames. Details for each part are given in the following subsections.

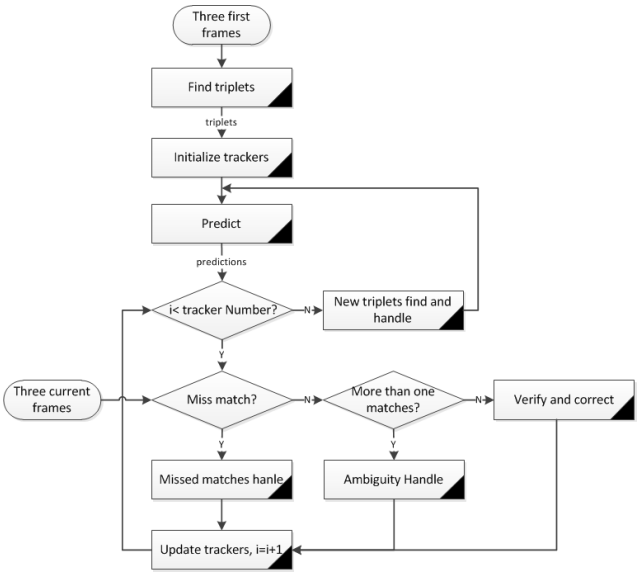

Figure 2: Flow chart of the overall strategy

Triplets. At time $t = 1$, initial triplets are obtained by matching and verifying. For each $\mathbf{d}^1_{n_1}$, the relative triplet $(\mathbf{d}^1_{n_1}, \mathbf{d}^2_*, \mathbf{d}^3_*)$ is obtained according to Section 3.1. In $\mathcal{I}^2, \mathcal{I}^3$, there may be still some detections not matched to any *triplet*, due to the occlusion in $\mathcal{I}^1$. The triplets for those detections are obtained by matching and verifying according to $\mathbf{d}^i_{n_i}$, for $i = 2, 3$. Thus, each triplet is related to one and only one object. One detection may belong to several triplets due to occlusion.

Tracking. Three trackers $\mathbf{T}^1_k, \mathbf{T}^2_k, \mathbf{T}^3_k$ (UKF) are initialized for the kth triplet (object). For $t > 1$, 2D tracking (Section 3.2) is operated separately by the trackers. New triplets are obtained by tracking only, if there is no ambiguity or missed match. But the new triplets are verified by checking projection consistency to identify wrong tracking results and correct them. If projection consistency is satisfied, the 3D position for the object is obtained by projecting the triplet onto 3D coordinates.

Correction. We assume that there is no more than one wrong tracking result. Cross projection consistency is conducted to find and correct a false tracking result. Let

$$dist(\mathbf{q}^j_{n_i}, \mathbf{d}^j_*) \quad < \quad \tau_2 \qquad (1)$$
$$\mathbf{d}^j_* \quad = \quad \min_{n_j}(dist(\mathbf{q}^j_{n_i}, \mathbf{d}^j_{n_j}) + (\mathbf{l}^j_{n_i})^{\mathbf{T}}\mathbf{d}^j_{n_j})$$

If $\mathbf{d}^j_*$ exists and it is not the same point in the matched triplet, then tracking in $\mathcal{I}^i$ is wrong, and should be corrected to be $\mathbf{d}^i_*$ instead, where τ_2 is the threshold parameter for projection consistency. The value of τ_2 depends on calibration accuracy.

If none of the three satisfy Equation (1), which means that two tracking results are false, then this trajectory is terminated.

Ambiguity Handling. If more than one detection is matched to a trajectory, as shown in Figure 1, the possible triplet configurations are verified with projection consistency to find the triplet that satisfies

$$dist(\mathbf{q}^i_{n_i}, \mathbf{d}^i_{n_i}) \quad < \quad \tau_2 \qquad (2)$$

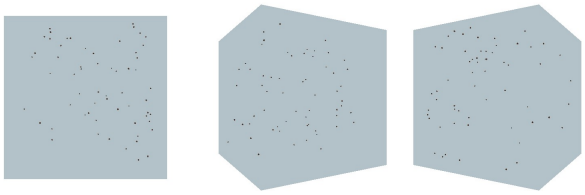

Figure 3: Screen shot of the simulator environment.

If more than one triplet satisfies Equation (2), the distance to the predicted point is taken into consideration. We take the one satisfying

$$d_*^i = \min_{n_i} dist(\mathbf{q}_{n_i}^i, \mathbf{d}_{n_i}^i) + dist(\mathbf{x}_{t|t-1}^p, \mathbf{d}_{n_i}^i) \quad (3)$$

as the valid one. If none of them satisfies Equation (2), the trajectory is terminated.

Missed Match Handling. If one tracker $\mathbf{T}^i$ misses the match as shown in Figure 1, $\mathbf{d}^i$ can be retrieved by matching the projection of the 3D coordinate reconstructed by the other two correct tracking results, satisfying Equation (1). If none of the them satisfies Equation (1), this trajectory is terminated.

New Objects. If there are untracked detections in any of the three current images after propagating all trackers, triplets corresponding to these detections are found and then new trackers are initialized.

4. 3D SIMULATOR

The swarm simulator is developed to generate all necessary images and matrices to calculate and validate the tracking algorithm. To guarantee realistic movement without restricting the simulator to a specific animal or object, two different random walk models are integrated. All parameters of these models can be set individually to simulate different behaviours. Examples for those parameters are target/chamber size, take-off probability or maximum flight speed.

The 2D positions are calculated from 3D positions using extrinsic and intrinsic matrices. The focal length is calculated employing the field of view of the simulated camera in x direction (f_x) using the equation

$$f = \frac{x}{2 * \tan(\frac{f_x}{2})}$$

Two different random walk models are available to simulate different behaviour. Both models are initialised with random locations and velocities (if not specified differently). The Gaussian model is commonly used to simulate particle swarms. It is based on the velocity equation

$$v_t^i = \Theta v_{t-1}^i + n_t,\ i = 1, 2, 3$$

where $n_t \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian noise with zero mean and σ standard deviation (independently chosen for each $i \in \{1, 2, 3\}$) added to the velocity vector and $\Theta \in [0, 1]$ is a parameter to control the smoothness. The location of the targets is then updated from the previous position by $x_t^i = x_{t-1}^i + v_t^i$ for all $i \in \{1, 2, 3\}$ and for each time step and projected to the image planes.

The second random walk model defines random time windows in which the velocity remains unchanged. For each time window, a specific velocity is again randomly calculated specified by the parameters given by the user. Interpreting the velocity as an offset, this offset is added to the current positions. In case of a wall contact, the movement speed is reduced by a constant factor and the probability to crawl or sit is used to simulate 2-dimensional movement in both random walk models.

In addition, both models support *negative geotaxis* (i.e. the tendency of fruit flies to orient themselves to the top [11]) by a custom probability to force the y-axes offset to be zero or positive.

To simulate the free flight of dozens of fruit flies, we adjusted all parameters based on ascertained measurements found in the literature.

5. EXPERIMENTS

We test the proposed method using simulated particle swarms generated by the 3D simulator described above. Compared with the first random walk model with smoothness control, used in [24], the second model is more irregular and realistic. Both models are employed in our experiments. Detections (of centroid coordinates) are obtained by adding 1-2 pixel noise to the ground truth. The world coordinate origin is set to be at the cube's centre. The poses of the cameras are set to be (0,0,8) (translation), 0° (rotation about the Y-axis) for Camera 1, (0,0,8), $-120°$ for Camera 2, and (0,0,8), 120° for Camera 3. A screen shot of three images from the simulator is shown in Fig. 3.

In the following experiments, the scene is captured with a frame rate of 150 fps and resolution 800×800. Furthermore, the size of the chamber and of the flies, as well as the movement parameters are set proportional to real-world values. The chamber is $20 \times 20 \times 20cm^3$ and each fly is represented by a sphere with a radius of $2mm$. Since the beam width of the field-of-view is 45°, we placed the cameras $800mm$ away from the cube's centre (compare to translation vector above). According to Marden et al. [14], the maximum flight speed is set to $0.8m/s$ whereas the crawling speed is reduced by the factor 0.1. Scenes containing different numbers of objects, from 20 to 100, are adopted to test the algorithm.

5.1 2D Tracks

Figure 4 (right) shows the track and position estimates of 40 objects in an image of Camera 1. The estimations (red dots) obtained from UKFs are following the tracks properly. Different colours correspond to different objects. Triple pro-

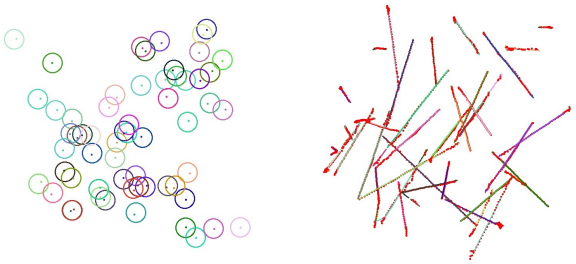

Figure 4: Search regions for 2D tracking and estimations for 40 objects. Left: circles denote the search regions for trackers defined by predictions from UKF and $\tau_1 = 25$; right: 2D tracks overlaid with estimations (red dots)

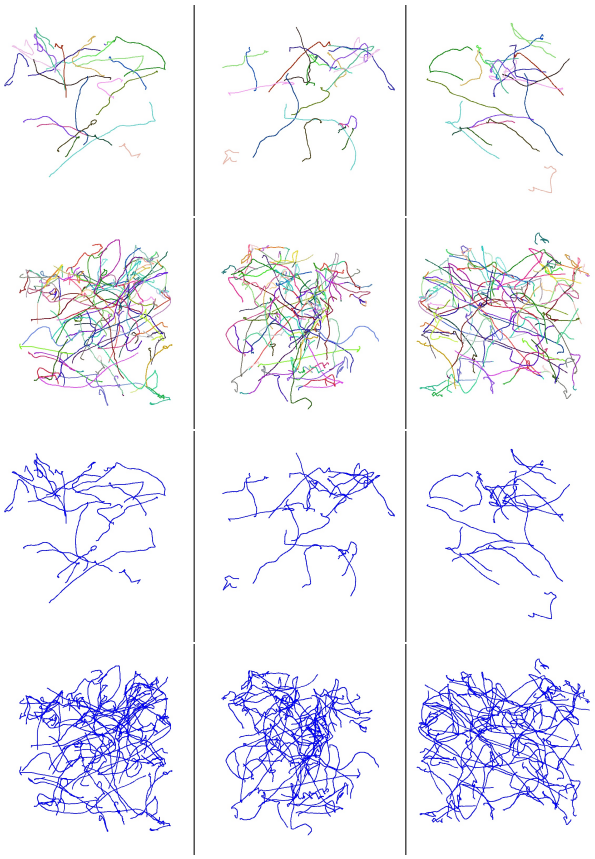

Figure 5: Triplet trajectory results tested for the random walk model with smooth control (20/80 objects). Top: triplet tracking results. Bottom: tracking results overlaid with ground truth (blue lines)

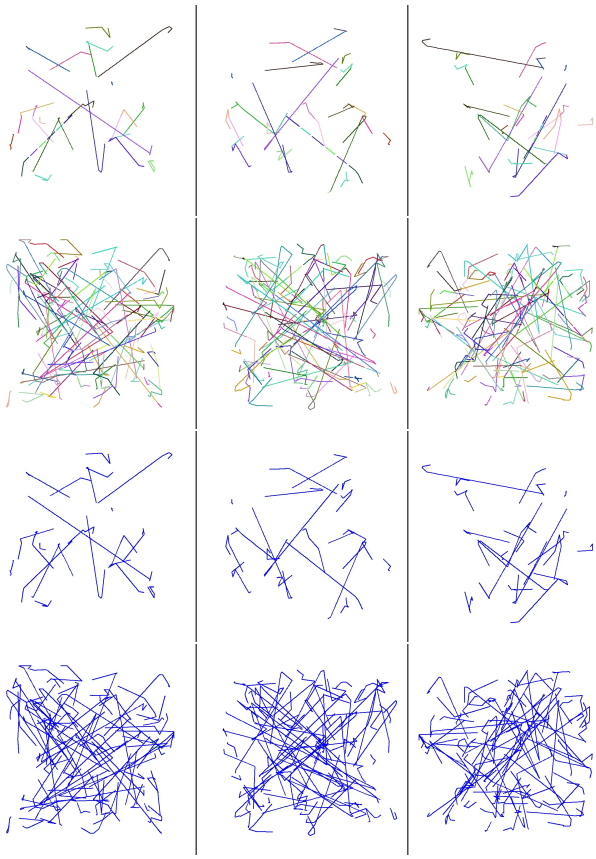

Figure 6: Triplet trajectory results on the second random walk model without smoothness control (20/80 objects). Top: triplet tracking results. Bottom: results overlaid with ground truth (blue lines)

jections of one object are shown in the same colour. In Figure 4 (left), the circles show the search regions for new triplets. Centres of circles are the predictions obtained from UKFs. The radius is related to the parameter τ_1. The larger τ_1, the less miss-matches, but the larger the processing time. At an early stage, a large τ_1 is recommended, as UKFs are initialized by zero velocity (no prior information is available). τ_1 is also related to the maximum velocity of the tracked objects and the frame rate.

5.2 3D Trajectories

The matched detections are connected to tracks in the 2D image plane. With two or three corresponding points, the 3D trajectories can be reconstructed. Tracking results of scenes containing 20 and 80 objects, with or without smoothness control, are shown in Figs. 5 and 6 respectively. The blue lines are ground truth; coloured lines denote tracking results with proposed method.

Table 1 shows evaluation results by listing *correspondence and association errors* defined as follows:

$$E_{ca} = \frac{N_c + N_a}{T}$$

Here, N_c and N_a denote the numbers of incorrect stereo matches (i.e. more than one element in one triplet is wrong) in all the frames, and the false temporal association between adjacent frames. For the number of frames we have $T = 334$. E_{ca} is similar to the RAE measure in [24] which includes the number of objects N in the denominator. With more objects in the scene, larger values of N_c, N_a may yield a small RAE, due to a large denominator. In the table, E_{ca} or RAE 1st, and E_{ca} or RAE 2nd denote results for first (Gaussian smoothness) and second random walk model, respectively.

By increasing the number of objects, more misses and false associations occur. If more than one candidate is specified by an UKF in the search region, then the one closest to the prediction might be a wrong association, which occurs in crowded scenes more frequently. Thus, N_a increases with the number of the objects. N_c is relatively small, but when dealing with real-world sequences, more noise (e.g., introduced by detection or calibration) may lead to larger N_c values, especially in crowded scenes. As we allow multiple tracks associated with one detection, occlusions will not make our method to fail. But, on the other hand, tracks may be merged or segmented in this case.

Results for the walking model with smoothness control are much better than for the one without, as the motion is more smooth between adjacent frames. However, the proposed

Table 1: Performance Measure with E_{ca} and RAE

N	20	40	60	80	100
E_{ca} 1st	0.001	0.012	0.012	0.245	0.235
E_{ca} 2nd	0.033	0.060	0.147	0.314	0.500
RAE 1st	0.000	0.001	0.001	0.002	0.001
RAE 2nd	0.001	0.001	0.001	0.002	0.003

method is able to track simulated fruit flies for both random models. Different to [24], we do not require specifying the number of objects for the algorithm.

6. CONCLUSIONS

In this paper, we address the 3D fruit fly trajectory reconstruction task in a simulated environment, with realistic parameters and noise set or added to the process. With ground truth available, the performance of the proposed algorithm is tested on scenes with different numbers of objects. The experimental results are promising as shown above. A remaining problem is that trajectories may merge together or may break into segments. Such undesired segmentation and merging of tracks will be further approached by tracking one triplet with one tracker, as the three elements are related to each other. Situations where objects are close to edges of the cube require extra future efforts. The formulated constraints also need to be tested further for noisy inputs, especially when applying the algorithm for tracking of real fruit flies. In a 3-camera system, three sets of epipolar constraints, and three sets of projection consistency constraints are available for finding triplets. At this time, we only use one epipolar and one projection consistency constraint.

Currently we collaborate with the Institute for Neuro and Behavioural Biology Münster designing a real world tracking setup. Three high speed and high resolution Basler cameras (*acA2040-180km*) are utilized to acquire real-world recordings with comparable temporal and spatial resolution. Since the synthetic camera matrices are equally constructed to real world calibration outputs (e.g. gathered by the OpenCV library), it should be easy to apply our algorithm to real world data.

7. ACKNOWLEDGMENTS

The authors thank *Gabriel Hartmann* for support regarding the unscented Kalman filter.

8. REFERENCES

[1] M. Betke, D. E. Hirsh, A. Bagchi, N. Hristov, N. C. Makris, and T. Kunz. Tracking large variable numbers of objects in clutter. In Proc. *CVPR*, pages 1–8, 2007.

[2] H. Buelthoff, T. Poggio, and C. Wehrhahn. 3-d analysis of the flight trajectories of flies (drosophila melanogaster). *Z. Naturforschung*, 35c:811–815, 1980.

[3] H. Dankert, L. Wang, E. D. Hoopfer, D. J. Anderson, and P. Perona. Automated monitoring and analysis of social behavior in drosophila. *Nature Methods*, 6:297–303, 2009.

[4] U. R. Dhond and J. K. Aggarwal. Structure from stereo – a review. *IEEE Trans. Systems Man Cybernetics*, 19:1489–1510, 1989.

[5] H. Du, D. Zou, and Y. Q. Chen. Relative epipolar motion of tracked features for correspondence in binocular stereo. In Proc. *ICCV*, pages 1–8, 2007.

[6] S. N. Fry, M. Bichselb, P. Müllera, and D. Robert. Tracking of flying insects using pan-tilt cameras. *Neurosci. Methods*, 101(1):59–67, August 2000.

[7] A. Gomez-Marin and M. Louis. Active sensation during orientation behavior in the drosophila larva: More sense than luck. *Current Opinion Neurobiology*, 22:208-215, 2011.

[8] D. Grover, J. Tower, and S. Tavaré. O fly, where art thou? *Royal Society*, 5:1181–1191, 2008.

[9] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2003.

[10] A. Y. Katsov and T. R. Clandinin. Motion processing streams in drosophila are behaviorally specialized. *Neuron*, 59:322–335, 2008.

[11] K. J. Kohlhoff, T. R. Jahn, D. A. Lomas, C. M. Dobson, D. C. Crowther, and M. Vendruscolo. The ifly tracking system for an automated locomotor and behavioral analysis of drosophila melanogaster. *Integr. Biology (Camb.)*, 3:755–760, 2011.

[12] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In Proc. *ECCV*, pages 8–40, 2002.

[13] S. Lahiri, K. Shen, M. Klein, A. Tang, E. Kane, M. Gershow, P. Garrity, and A. D. T. Samuel. Two alternating motor programs drive navigation in drosophila larva. *PloS ONE*, 6(8):e23180, 2011.

[14] J. H. Marden, M. R. Wolf, and K. E. Weber. Aerial performance of Drosophila melanogaster from populations selected for upwind flight ability. *Experimental Biology*, 200:2747–2755,1997.

[15] J.-R. Martin. A portrait of locomotor behaviour in drosophila determined by a video-tracking paradigm. *Behavioral Processes*, 67:207–219, 2004.

[16] T. A. Ofstad, C. S. Zuker, and M. B. Reiser. Visual place learning in drosophila melanogaster. *Nature*, 474(7350):204–207, 2011.

[17] F. Pereira, H. Stuer, E. C. Graff, and M. Gharib. Two-frame 3d particle tracking. *Measurement Science Technology*, 17:1–8, 2006.

[18] I. Schmidt, S. Thomas, P. Kain, B. Risse, E. Naffin, and C. Klämbt. Kinesin heavy chain function in drosophila glial cells controls neuronal activity. *Neuroscience*, 32:7466–7476, 2012.

[19] M. B. Sokolowski. Drosophila: Genetics meets behavior. *Nat. Rev. Genet.*, 2:879–890, 2001.

[20] A. D. Straw, K. Branson, T. R. Neumann, and M. H. Dickinson. Multi-camera real-time three-dimensional tracking of multiple flying animals. *Royal Society*, 8:395–409, 2011.

[21] L. F. Tammero and M. H. Dickinson. The influence of visual landscape on the free flight behavior of the fruit fly drosophila melanogaster. *Experimental Biology*, 205:327–343, 2002.

[22] E. A. Wan and R. van der Menve. The unscented Kalman filter for nonlinear estimation. In Proc. *Adaptive Systems Signal Processing Communications Control*, pages 153–158, 2000.

[23] H. S. Wu, Q. Zhao, D. Zou, and Y. Q. Chen. Acquiring 3d motion trajectories of large numbers of swarming animals. In Proc. *ICCV Workshop*, pages 593–600, 2009.

[24] D. Zou, Q. Zhao, H. S. Wu, and Y. Q. Chen. Reconstructing 3d motion trajectories of particle swarms by global correspondence selection. In Proc. *ICCV*, pages 1578–1585, 2009.